An Adaptable Deep Learning System for Optical Character Verification in Retail Food Packaging

Fabio De Sousa Ribeiro and Francesco Caliva
School of Computer Science
MLearn Group
University of Lincoln
LN67TS, Lincoln
United Kingdom
{fdesousaribeiro, fcaliva}@lincoln.ac.uk

Mark Swainson and Kjartan Gudmundsson
National Centre for Food Manufacturing
University of Lincoln
Holbeach Technology Park
PE127PT, Holbeach
United Kingdom
{mswainson, kgudmundsson}@lincoln.ac.uk

Georgios Leontidis and Stefanos Kollias
School of Computer Science
MLearn Group
University of Lincoln
LN67TS, Lincoln,
United Kingdom
{gleontidis, skollias}@lincoln.ac.uk

***Abstract*—Retail food packages contain various types of information such as food name, ingredients list and *use by* dates. Such information is critical to ensure proper distribution of products to the market and eliminate health risks due to erroneous mislabelling. The latter is considerably detrimental to both consumers and suppliers alike. In this paper, an adaptable deep learning based system is proposed and tested across various possible scenarios: *a)* for the identification of blurry images and/or missing information from food packaging photos. These were captured during the validation process in supply chains; *b)* for deep neural network adaptation. This was achieved through a novel methodology that utilises facets of the same convolutional neural network architecture. Latent variables were extracted from different datasets and used as input into a *k*-means clustering and *k*-nearest neighbour classification algorithm, to compute a new set of centroids which better adapts to the target dataset's distribution. Furthermore, visualisation and analysis of network adaptation provides insight into how higher accuracy was achieved when compared to the original deep neural network. The proposed system performed very well in the conducted experiments, showing that it can be deployed in real-world supply chains, for automating the verification process, cutting down costs and eliminating errors that could prove risky for public health.**

***Index Terms*—deep learning, convolutional neural networks, clustering, trained representations, adaptation, optical character verification, retail food packages**

I. Introduction

Systems of national and global food supply are often complex and multifaceted, characterised by multiple stages of processing and distribution. In the European Union, food production is the largest manufacturing sector accounting for 13.3% of the total EU-28 manufacturing sector with a reported turnover of 945 billion [1]. Whilst food availability is a primary concern in developing nations and food quality/value a focal point in more affluent societies, food safety is a requirement that is common across all food supply chains. Food safety in the sector is typically underpinned by food science and technology and assured by a combination of operational control systems and procedures including Good Manufacturing Practice (GMP) and Hazard Analysis & Critical Control Point (HACCP) [2]. Pre-packaged food products, which are incorrectly labelled (e.g. bearing an incorrect or illegible *use by* date, see Figures 1 and 2), result in product recalls as the fault/issue could cause a food safety incident such as food poisoning due to the consumption of the product which is past its safe *use by* date for consumption. These recalls are usually at very high financial cost to food manufacturers, often compromising their reputation. Recurring root causes for issues/mistakes resulting in label faults on food packaging are many and varied. They are mainly comprised of but are not limited to human error and equipment faults. For example, a label printer on a production line can break down and the line carries on running. The faulty packaging therefore needs to be identified and the production line stopped.

(a) (b)

Fig. 1. Example of common food packaging labels to be targeted by the proposed system. The labels show *use by dates* which are of highest interest and the additional traceability codes (e.g. L 012B 16 16) are also present.

A common approach is to read and verify the *use by* dates on packaging labels. Usually, a human operator performs this check by either manually picking a pack from the line for inspection or verifying it through an image captured of the pack. However, these methods create mundane and repetitive tasks and therefore place the operator in an error-prone working environment. They also struggle to offer statistically

(a) (b) (c) (d) (e)

Fig. 2. The above images depict an example per category of data annotation during pre-processing of the datasets. a: Complete Date (day and month visible). b: Partial Date (no day visible). c: Partial Date (no month visible). d: Unreadable e: No date (neither day or month visible).

significant data for analysis by the supplier. For example, checking the correctness of a pack once every 5 minutes (when the line is running at 100 packs per minute) enhances the risk of missing a fault across the other 499 packs processed during the check interval. Another common approach to control date codes is to use Optical Character Verification (OCV). This involves a supervisory system holding the correct date code string and transferring it to both the printer and the vision system. The latter will then verify its read and depending on the result, actions are taken. However, OCV systems rely on consistency in date code format, packaging and camera view angle. This consistency tends to be hard to achieve in the food & drink manufacturing environment and therefore there is a great need for a more robust solution. In this work an adaptive deep neural network approach for OCV systems is proposed. The system appropriates the transfer of knowledge across multiple domains and adapts it to the problem at hand. Section II presents related work in deep learning and applications. Section III describes the datasets and data pre-processing used in this work. Section IV describes the novel proposed approach, focusing on the convolutional neural network architecture (CNN), transfer learning and adaptation strategy. Section V presents the experimental results obtained when applying the developed deep neural architectures to real food packaging data. Conclusions and further work is described in Section VI.

II. Related Work

Recent successes of Deep Learning (DL) have significantly boosted its popularity. This has resulted in a manifold of DNN applications in a variety of domains, including Computer Vision, Signal and Natural Language Processing ([3]–[6]). The increased availability of computation resources has resulted in new advanced algorithms able in some cases to perform better than humans. However, for particular problems, the amount of data available appears to be a limitation, and overfitting its main drawback. Companies like Google whose data availability is abundant do not face such problems, as overfitting appears to be a big issue when dealing with relatively small datasets. To reduce overfitting, a number of strategic choices can be made. For instance, Dropout training has been widely adopted with very good levels of success ([7], [8]). Furthermore, scarcity of data can further be mitigated with data augmentation [9]. Transfer learning (TL) has also proven to be very effective in adapting knowledge related to data from different distributions and domains [10]. The knowledge is transferred in the form of feature representation of learned models and is utilised to solve differing tasks [11]. Recently, domain adaptation - a special case of TL - has also become a very popular practice in computer vision. Such algorithms focus on minimising the prediction error on a target dataset. Generally, DNNs are trained with labelled training data, more commonly known as supervised learning. Consequently, domain adaptation involves classification on a dataset from a different distribution from the one used during training, regardless of the availability of sufficient labels. TL and in particular domain adaptation have found application in several fields, which relate to robotics, object detection, video analysis, medical imaging and other fields where data availability is limited. Nevertheless, there exist many adversities such as the potential absence of labels or the lack of data to train complex DL models ([12], [13]).

In recent research, DL has been employed in various ways to perform text detection. Huang *et al.* [14] first identified candidate text regions and then utilised deep CNN to discard false positives. Jaderberg *et al.* [15] combined aggregated channel features with CNN to spot text in photographs. Tian *et al.* [16] used a combination of vertical anchors with a joint Convolutional-Recurrent Neural Network (C-RNN) to detect horizontal lines of text. However, these methods are comprised of multiple components all requiring appropriate tuning, which contributes to the overall processing time needed per sample. The proposed approach is mainly comprised of transfer learning and domain adaptation techniques. In particular, a convolutional neural Architecture [17], with pre-trained weights was employed to initiate transfer learning. Furthermore, Dropout training along with data augmentation were implemented to reduce the risk of overfitting on the specific data collected. Lastly, a novel domain adaptation framework is proposed and results of the experimental study are given in the context of the current OCV problem.

III. Data Pre-processing

Figure 3 is exemplary of the dataset utilised. As can be inferred, the degree of difficulty of an *at-first-glance* recognition of the *use by* date differs from one case to another.

TABLE I
NUMBER OF IMAGES PER DATASET

Label (DD/MM)	Dataset 1	Dataset 2	Dataset 3
Missing/Missing	375	3715	0
Missing/Complete	59	68	16
Complete/Missing	10	39	0
Complete/Complete	645	2847	1138
Unreadable	315	46	0

Fig. 3. Demonstration of the variability of the interpretability of the labels. On the left hand side it is relatively simple to recognise the *use by* date. On the right side it is not straightforward.

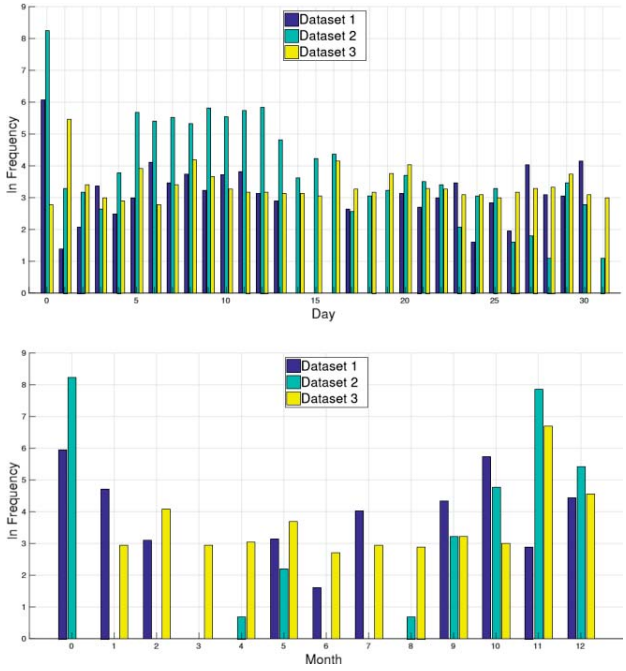

Fig. 4. **Top:** Frequency (ln scale) of appearance per day in food package *use by* dates. **Bottom:** Frequency (ln scale) of appearance per month in food package *use by* dates.

Three datasets comprising images showing food package labels were collected by a leading food company and provided to us for research purposes. The three sets included 1404, 6739 and 1154 captured images respectively.

In order to produce trainable datasets, the images were first manually annotated with respect to (w.r.t) the presence of *use by* dates, and lack thereof. Specific sorting criteria were employed regarding the grouping of images based on what date information was missing. In the case of unreadable images, meaning that upon inspection a date was not discernible from the rest of the background, due - but not limited to heavy distortion, non-homogeneous illumination or blur, these were then set aside in a separate category. Otherwise, images in which either day or month, or both were missing, were considered as incomplete, and subsequently grouped into their own category. Lastly, images of good quality, reporting the date including both the day and month, were considered as good candidates for OCV.

In summary, three different sets of images were annotated conforming to the critera mentioned above to form 5 categories: complete dates, missing day, missing month, no date and unreadable (Table I). Having annotated all three sets of images, it was possible to calculate the statistics (see Figure 4) regarding the frequency of specific dates within each dataset, and thus devise a methodology for conducting experiments with balanced sets of classes. The scarcity of training data appeared to be the major limitation for the dataset at disposal. Moreover, by inspecting the images with partially missing data, it was observed that most of them were photographs of package labels which had been folded at crucial points, included photographic glare, digits fainting over time, or included human made occlusions. To overcome the inherent lack of training examples and alleviate class imbalance, for one out of the three datasets, 577 images were augmented in a similar manner to existing human intervened ones. Additionally, regions with visible *use by* dates were cropped to devise experiments which offered insight into the effect of background noise on the classification (see Figure 5).

IV. THE PROPOSED APPROACH

To address the major aforementioned food production challenges, a methodology was devised to develop a robust machine vision solution for date-code verification. Concretely, in this study a global approach to OCV is proposed. This is achieved through leveraging the feature extraction power inherent to Deep Neural Networks. Furthermore, by utilising adaptation strategies such as transfer learning, *k*-means clustering and *k*-Nearest Neighbour classification of representations extracted from the CNN, it was possible to compensate for the limited amount of training data available. The implementation was based on MATLAB [18], Keras deep learning framework [19] and Tensorflow numerical computation library [20]. The experiments were conducted using a server with an Intel Xeon(R) E5-2620 v4 CPU, eight GPUs and 96GB of RAM.

A. Convolutional Neural Networks

CNN architectures ([17], [21]) are neural networks that comprise filtering layers, in which a number of affine transformation and subsequent non-linearity are applied to an input vector. It is common that CNN, particularly when image based, use pooling layers to summarise the activation of multiple adjacent filters within a single response, and also to add robustness to the model against input translations. Usual pre-trained CNN architectures take as input three channelled

Fig. 5. On the left hand side an example is given showing an image taken from the third dataset. On the right hand side, we depict the cropped ROI including the respective *use by* date. Both images were used for classification and the results between cropped vs non-cropped were compared.

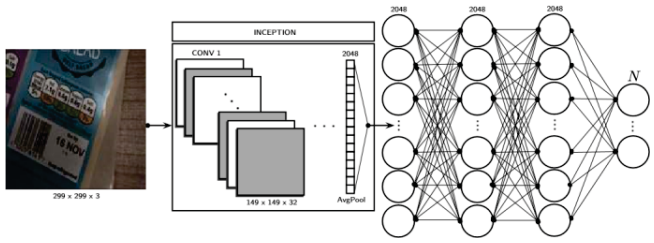

Fig. 6. Depiction of the classification architecture. From left to right, input images were resized to 299×299×3 to accommodate the CNN's convolutional layer parameters and arithmetic. There exist 2 hidden layers with 2048 units each and ReLu activations. The number of units *N* in the Softmax layer was adjusted as per the number of classes being classified in different experiments.

images and through a series of volume-wise convolutions and feature routing, are capable of selecting the optimal features necessary for the classification of particular objects. This in turn eliminates the need for hands-on feature engineering approaches, as is the custom in classical Machine Learning and Computer Vision.

B. Transfer Learning

It was of particular interest to conduct transfer learning and assess the adaptability of pre-trained CNN weights on the current food datasets [17]. Specifically, each image was fed through a previously trained network on the Imagenet dataset, up to the last pooling layer, where a 2048 dimensional vector representation of each instance was extracted. The 2048 dimensional vectors then became the input to a new series of fully-connected layers and a final Softmax layer able to predict N classes depending on the experiment being conducted (see Figure 6). Softmax, presented in (1) is a function $\sigma(\vec{x})$ which converts an N-dimensional vector $\vec{x}$ of arbitrary real values to a N-dimensional vector of real values in the range [0, 1] summing to 1.

$$\sigma(x_j) = \frac{e^{x_j}}{\sum_{i=1}^{N} e^{x_i}} \; for \; j = 1, \dots N \tag{1}$$

In order to optimise the training performance of the new fully-connected layer network, a series of architectural decisions were made empirically. The best performances were achieved with a fully-connected network consisting of two 2048 unit hidden layers with Rectified Linear Unit (ReLU) activation function.

$$ReLU :\rightarrow f(x) = \max(0, x) \tag{2}$$

The risk of overfitting rises as the number of parameters increases w.r.t number of training examples. Due to the limited amount of training data available for experimentation, it is unfeasible to train state-of-the-art models from scratch. Therefore, it was paramount to introduce an effective regulariser in the new network as well as to adapt previously learned low-level features by way of transfer learning. As of the time of this writing, the most effective regularisation technique is Dropout [8]. In practice, to preserve more information in the input layer $l^{[0]}$ (of L total layers) in the network and thus aid learning, the probability P of keeping a given neuron n (not setting it to 0) was as per the following schema.

$$l^{[i]} = \begin{cases} P(n) = 0.8 & if \;\; i = 0 \\ P(n) = 0.5 & otherwise \end{cases} \tag{3}$$

In view of the unbalance present among the various classes, it was beneficial to use weighted cross entropy as a loss function (4). In (4), ω_j is a weight coefficient computed for the j^{th} of all classes J as a function of the proportion of instances N_j compared to the most densely populated class (5). During training, ω encourages the model to focus on under-represented classes.

$$L(\hat{x}, x) \; = \; -(\omega_j x \log(\hat{x}) \; + \; (1 \; - \; x) \log(1 \; - \; \hat{x})) \tag{4}$$

$$\omega_j \; = \; \frac{\max(\{N_i\}_{i=[1:J]})}{N_j} \tag{5}$$

In the case if multiclass classification, where $J > 2$, the weighted cross entropy loss function is defined as

$$L(\hat{x}, x) \; = \; -\sum_{j=1}^{J} \omega x \log(\hat{x}) \tag{6}$$

C. Adapting Trained CNN Architectures

A major issue spanning the three datasets was the variability in the captured images characteristics. Particularly, the first two datasets are comprised of grey-scale images and the third of higher resolution colour images. Moreover, different background contexts were present within the three datasets, regarding the amount of textual and digit information. This variability made the reuse of a DNN trained on one dataset, for classifying the data of another, very difficult as it performed poorly. Fundamentally, this is because each dataset comes from a different distribution, as the images were taken by different people, with different cameras and at differing supplier locations. With limited data available, even the use of transfer learning among different environments and datasets can be ineffective, since there are generally few data in the new environments to successfully retrain a network. To overcome the challenges of both very

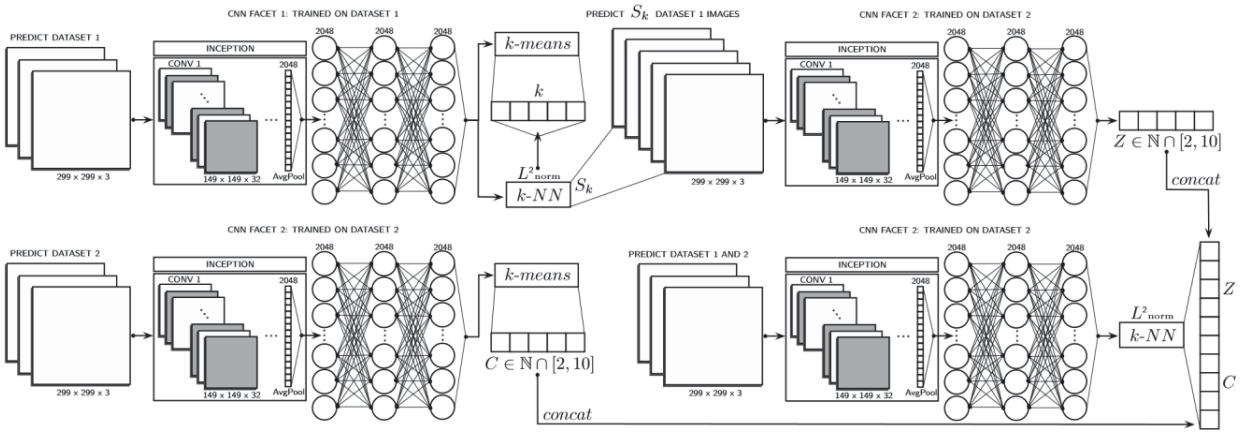

Fig. 7. Illustration of the multiple CNN Facet adaptation framework proposed. The CNN's architectural details are as previously described in Figure 6.

limited training data and high variability among the given datasets, we demonstrate the possibility of designing a new facet of the same CNN architecture, for learning each considered problem associated with different datasets. This focuses on: *i)* detecting bad image capturing conditions; *ii)* detecting missing dates (*i.e.* either day and/or month of *use by* date); *iii)* showing the ability to recognise day and/or month of an existing *use by* date. However, the derived CNN facets are inherently different and from the user's perspective, it is not evident which network out of existing ones, is most appropriate to solve the problem at hand.

Therefore in this paper, we propose a novel methodology for visualising and analysing variabilities between distributions and attempt to adapt information from one problem to another (Figure 7). Given two facets of the same CNN architecture, trained with two different datasets and targeting the same classification/recognition task, we propose the use of latent variables extracted from each trained network for adaptation. Visualisation and analysis of these latent variables highlights the differences in performance of a trained network, when it is inferred on a dataset coming from a different distribution. Specifically, by focusing on the $N^{[L-1]}$-dimensional output of the last fully-connected layer $L-1$ (where $L-1$ precedes the Softmax layer L and $N^{[L-1]}$ is the number of neurons in $L-1$), it can be observed that upon successful CNN training, the latent variables encode all valuable information necessary to perform classification. With that in mind, a *k*-means and *k*NN combined methodology was devised to cluster these representations into *k* clusters for each network. Formally, given extracted $N^{[L-1]}$-dimensional activations ($\vec{x}_1, \vec{x}_2, ..., \vec{x}_n$), from the last fully-connected layer *L*, as latent variable representations of *n* total input images, the objective function in (7) clusters them into *k* sets $C = \{C_1, C_2, ..., C_k\}$ as to minimise within-cluster L^2 norms.

$$arg\min_{C} \sum_{i=1}^{k} \sum_{x \in C_i} ||x - \mu_i||^2 \tag{7}$$

Clustering is performed separately on extracted activations from the two networks (CNN1 trained on dataset 1 and CNN2 trained on dataset 2). Firstly, *k* centroids $C = \{C_1, C_2, ..., C_k\}$ are computed from clustering CNN2 trained on dataset 2 representations, and set aside. Then we compute a further *k* centroids from clustering CNN1 trained on dataset 1 representations and perform 1-NN for each of the *k* centroids w.r.t dataset 1 images. This procedure produces a set of images $S = \{S_1, S_2, ..., S_k\}$. The next step is to forward propagate *S* images through a network (CNN2) trained on a different distribution (dataset 2) to obtain a new set of centroids $Z = \{Z_1, Z_2, ..., Z_k\}$, where *S* dataset 1 images are considered an approximation of the *k* centroids previously obtained through clustering of dataset 1 representations. An important detail to mention, is that the cluster class label is given by the mode *j* class (*J* total classes) of data points assigned to that cluster.

$$C_i^j = \max_{j \in J} |C_i \cap j| \tag{8}$$

In summary, our adaptation approach is based on concatenating the respective *C* and *Z* cluster centroids in a new set of centroids $A = \{C_1, ..., C_k, Z_1, ..., Z_k\}$, deriving an augmented cluster representation which includes knowledge from both facets of the trained networks (CNN1 & CNN2). Since the different datasets will be closer to the cluster centroids extracted from the corresponding CNN with which they were trained, we demonstrate that they can be classified in the correct category using nearest neighbour (*k*-NN) classification w.r.t to the augmented cluster centroids (*A*). Lastly, given *A*, the last procedure consists of iteratively excluding the cluster centroid in *A* which achieves the lowest classification accuracy, and re-evaluating until the performance stops improving.

V. Experimental Study

Four sets of experiments were conducted and the obtained results are reported in Table II.

The goal of the first experiment was to establish a baseline for images that would be classified as acceptable according to human standards. As explained in section I, some of the implications of such an automated system include the elimination of tedious manual labour, reduced human error in

TABLE II
SETTINGS AND RESULTS OF THE EXPERIMENTS

Experiment	Dataset	# Images	Accuracy (%)
Complete dates vs Unreadable	1	645 vs 645	90.1
Complete dates vs Partial dates	1	645 vs 444	89.3
	2	2847 vs 2847	96.8
	3	577 vs 577	85.8
No date vs Partial & Complete	1	714 vs 375	94.8
	2	2954 vs 2954	96.2
	3	199 vs 199	85.8
Digit classification	2	381 vs 381 vs 381	92.7
	3	55 vs 67 vs 63 vs 61	90
	3	55 vs 67 vs 63 vs 61	95.7

package management and routing, increased speed and productivity, while providing statistically significant correctness rates. For the most part, images which were annotated as unreadable contained some form of heavy distortion such as photographic glare or blur, which rendered dates indiscernible from the background. Consequently, these must be filtered from the set of acceptable images and not be considered for further OCV processing. The second experiment aimed at distinguishing between acceptable and not-acceptable, missing dates. For this purpose, it was imperative to ensure that partial dates were not accepted by this process. This meant that the absence of either day or month digits in a *use by* date, had to be identified and filtered from the set of acceptable images. The challenge arose when one or both day/month sections were only partially obscured and thus may have been identified as acceptable by an advanced recognition system, but not by a human. Moreover, in many cases the partially missing date digits/letters could be present elsewhere in the image. Misclassification of these occurrences would lead to partial dates being processed as acceptable and affect the accuracy of the system.

In the third experiment, a global OCV approach was studied, targeting at verifying specific digits and letters, for which a sufficient number of samples existed in the used datasets. Successful text recognition systems typically begin with the detection of text presence within a given image, followed by a segmentation or localisation of the desired region-of interest in order to perform classification thereafter. All three datasets were comprised of images including text variations, such as differing fonts, sizes, colours and orientation angles, all in close proximity to the desired date. This posed a major challenge in the localisation of the desired date within a given image, as it required all other types of text in which the same digits or letters would appear, to be ignored. In this experiment we assess the efficacy of the CNN's feature extraction capabilities for automatic localisation and verification of specific digits/letters in the images.

In the fourth set of experiments we examined the ability of the proposed adaptation approach to improve the classification performance of a deep neural network trained with one dataset when it is used to process another one of the provided datasets.

Fig. 8. Example of data augmentation. On the left hand side, an image found within the dataset, on the right hand side an augmented image.

Fig. 9. Example of OCV classification. On the left hand side, a label whose *use by* date is relatively easy to be identified. On the right hand side, the proposed algorithm was able to recognise (in the experiment 5 vs 8) the correct day (8) despite the presence of several 5's across the image.

A. Complete Dates vs Unreadable

As a result of the annotation criteria explained in section III, the appearance of unreadable images was especially prominent in the first of the three datasets. Conversely, the average image quality of the second and third sets was higher and therefore they were not considered in this experiment. Moreover, the first dataset contained images from seven different locations, and as such, there were at least seven different types of food packaging. To devise a balanced experiment, images from all locations were combined and then categorised into 2 classes: Complete Dates and Unreadable. The images were then fed through the CNN network described in the former Section and their related 2048-dimensional vector representations were extracted. Utilising these vector representations, the fully-connected part of the network (see section: IV) was trained w.r.t to the class labels. As reported in Table II, a very satisfactory 90.1% accuracy was achieved over all seven different locations.

B. Complete Dates vs Partial Dates

In this experiment, the previously discussed issue of misleading partial dates was addressed. The second dataset was the largest, containing approximately 50% of examples with partial, or missing, dates. Images missing the day/month or both were assigned to one class and Complete Dates to the other. Utilising this newly categorised dataset, a similar classification work-flow described in the first experiment was employed and a new series of experiments were conducted. As reported in Table II, an excellent accuracy of 96.8% was achieved in identifying whether the best before data was

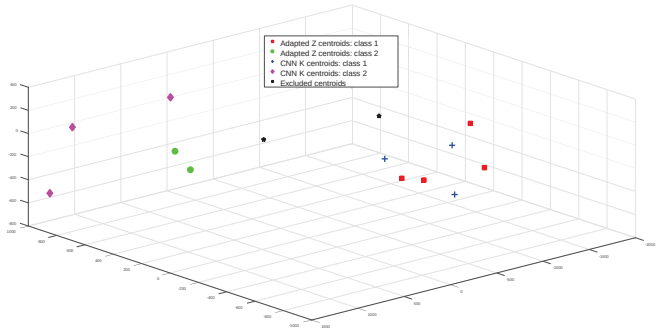

Fig. 10. t-SNE visualisation of the cluster centroids used to achieve 76.4% adaptation accuracy. As shown, clear separation was achieved between the per class cluster centroids computed from CNN2 representations on Dataset 2, and the new Dataset 1 adapted centroids following the *k*-means and *k*-NN proposed approach.

complete or not. Similarly, although using significantly fewer training examples, a performance of 94.8% was achieved when applying the same procedure to the first dataset.

In extension, a similar experiment was conducted on the third dataset, which includes images of higher quality. However, a very small number of missing value examples was included in this dataset. Through data augmentation, we produced a larger set of Partial Dates, see example in Figure 8. On this synthetic set, a good accuracy of 85.8% was also achieved.

Lastly, a small variation of this experiment was conducted in order to assess how well the network can identify the presence of any type of date, be it complete or partial, versus the absence of a date altogether (Partial+Complete Date vs No Date). This experiment offered insight into how well the network can produce inferred localisation of dates, as it must learn to filter out the abundant non-date related text within images. Table II shows that good accuracies were achieved across all three datasets, with the best case of 96.2% date presence detection on the second dataset.

C. Date Verification

Given that almost all images in the third dataset contained Complete Dates, it was possible to conduct the global approach to OCV previously explained in section V. Despite the small number of training examples (1138) and limited possible class combinations, four classes were formed: the first class comprised dates containing the number five in the day section; the second contained examples with the number eight; the third contained the number sixteen; the fourth contained the number twenty. With this labelled dataset, an accuracy of 90% was achieved in the global verification of the aforementioned digits.

In addition, to quantitatively assess the impact of background noise in the global OCV approach. Regions-of-interest containing only *best before* dates in the third dataset were cropped to form a new dataset for classification. Intuitively, the same work-flow, architecture parameters and train/test set split were utilised as in the previous approach and a higher classification accuracy of 95.7% was achieved, compared to the respective accuracy of 90% obtained in the previous experiment. This shows the potential of cropping the date area for obtaining better OCV results.

TABLE III
RESULTS OF THE ADAPTATION EXPERIMENTS.

	Accuracy (%)	
	CNN 2	Proposed Approach
Dataset 1	63.8	76.4
Dataset 2	95.9	97.1

Another global OCV based experiment was conducted on the second dataset, on detecting between the months of October and November classes, that were mentioned in sufficient number of respective data. Moreover, an equal sized portion of missing data images was also selected to form a third class and ensure class balance. The resulting accuracy achieved was also high, equal to 92.7%, showing that the network was able to distinguish between a date containing either the month of October, November or no date at all.

In reflection of these results, it is important to remember the great variety of text and numbers included in each image. Despite this fact and limited training examples, the networks were able to automatically recognise the importance of specific digits and their respective locations, whilst ignoring the same or other digits located in close proximity. Figure 9 is exemplary of this phenomenon.

D. Network Adaptation Results

The CNN architectures proved to be very accurate in solving the missing/complete dates problem. Subsequently, it was explored whether the respective trained networks were suitable to carry out the network adaptation approach (see Section IV-C). Table III provides the results obtained when implementing the proposed methodology. We consider the CNN2 trained with dataset 2, obtaining an accuracy of 95.9%. We test its performance on the first dataset and find a lower accuracy of only 63.8%. Then we applied the proposed approach in this experiment. Initially the described CNN representations of the last fully-connected layer when trained with dataset 1 were extracted for the trained CNN. These representations are 2048 dimensional vectors. As explained in more greater detail in section IV-C, these representations were clustered by way of a *k*-means algorithm with $k \in \mathbb{N} \cap [2,10]$. Once the cluster centroids were computed, a nearest neighbour algorithm was employed to evaluate the presence of any correlation among the two performed classifications (one for each available dataset). In particular, each of the centroids identified for the fist dataset were merged with respective cluster centroids extracted from the CNN trained with the second dataset, both forming a set of centroids to be considered, using a nearest neighbour algorithm to classify all data in the first dataset. Figure 10 depicts a 3-D visualisation of the cluster centroids, for $k = 7$ for both datasets (14 in total). Squares (Red) and (Blue) crosses denote the centroids corresponding to the complete date class in the first and second datasets respectively. (Green) circles and (Pink) diamonds are the centroids in the missing date category and the (Black) stars indicate the centroids not used in the final classification as per the centroid exclusion policy explained

previously in section IV-C. Having merged the 14 centroid representations, we generated a combined facet of all the knowledge extracted from both trained CNNs. By implementing this procedure, the testing performance presented in Table III was achieved. The testing performance improved the accuracy obtained previously to 76.4% on dataset 1 and even slightly improved the accuracy originally obtained on dataset 2 to 97.1%.

VI. Conclusion and Future Work

This paper proposed an adaptive deep learning framework for Optical Character Verification. The system aimed to automise the identification of *use by* dates and it was tested on a food packaging labels dataset. The proposed solution was based on the use and adaptation of convolutional neural network facets. Given the scarcity of data available, CNN extracted representations were first clustered using a *k*-means algorithm, followed by a *k*-nearest neighbour adaptation approach to combine centroids computed for both CNNs. By doing this, better separation and adaptation was achieved when classifying representations learned by another CNN on a similar problem, whilst using a different dataset.

The OCV with Deep Learning technologies developed in this paper can enable far greater control over the accuracy and legibility of critical *use by* dates and also key trace-ability information in food and drink manufacturing operations, resulting in significantly increased food safety and compliance with related legislation. The technology developed may also provide far wider options for advancement of food package control including the confirmation of allergen labelling present and correct on pack labels; confirmation that the right packaging has been used for each food pack and that the sales scanning bar code on each pack is readable and correct. Additionally it is necessary to perform quality Control Vision Inspection checking that the product in the pack is visually correct. The availability of comprehensive and specific food product information and the ability to very effectively recall if required will in the future become a process that takes minutes rather than days [22]. Our future work will extend the experimental study on a larger dataset with the goal to create a fully-automated OCV system.

Acknowledgment

The research presented in this paper has received funding from EPSRC (Reference number EP/R005524/1) and Innovate UK (Reference number 102908), in collaboration with the Olympus Automation Limited Company, for the project Automated Robotic Food Manufacturing System. Francesco Caliva' is funded by the aforementioned project. The authors´ would like to thank Mr. George Marandianos for manually annotating all datasets used in this study.

References

[1] Manufacturing statistics - nace rev. 2. http://ec.europa.eu/eurostat/statistics-explained/index.php/Manufacturing_statistics_-_NACE_Rev._2. Accessed: 14-01-2018.

[2] Codex alimentarius commission basic food hygiene texts (fourth edition, 2009). http://www.fao.org/docrep/012/a1552e/a1552e00.htm. Accessed: 01-02-2018.

[3] Dimitrios Kollias, Athanasios Tagaris, Andreas Stafylopatis, Stefanos Kollias, and Georgios Tagaris. Deep neural architectures for prediction in healthcare. *Complex & Intelligent Systems*, pages 1–13, 2018.

[4] Chen Sun, Abhinav Shrivastava, Saurabh Singh, and Abhinav Gupta. Revisiting unreasonable effectiveness of data in deep learning era. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 843–852. IEEE, 2017.

[5] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. How transferable are features in deep neural networks? In *Advances in neural information processing systems*, pages 3320–3328, 2014.

[6] Francesco Caliva, Fabio De Sousa Ribeiro, Antonios Mylonakis, Christophe Demaziere, Paolo Vinai, Georgios Leontidis, and Stefanos Kollias. A deep learning approach to anomaly detection in nuclear reactors. In *2018 International Joint Conference on Neural Networks (IJCNN)*, 2018.

[7] Geoffrey E Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan R Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*, 2012.

[8] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958, 2014.

[9] Martin A Tanner and Wing Hung Wong. The calculation of posterior distributions by data augmentation. *Journal of the American statistical Association*, 82(398):528–540, 1987.

[10] Hemanth Venkateswara, Shayok Chakraborty, and Sethuraman Panchanathan. Deep-learning systems for domain adaptation in computer vision: Learning transferable feature representations. *IEEE Signal Processing Magazine*, 34(6):117–129, 2017.

[11] Yoshua Bengio. Deep learning of representations for unsupervised and transfer learning. In *Proceedings of ICML Workshop on Unsupervised and Transfer Learning*, pages 17–36, 2012.

[12] John Blitzer, Ryan McDonald, and Fernando Pereira. Domain adaptation with structural correspondence learning. In *Proceedings of the 2006 conference on empirical methods in natural language processing*, pages 120–128. Association for Computational Linguistics, 2006.

[13] Dimitrios Kollias, Miao Yu, Athanasios Tagaris, Georgios Leontidis, Andreas Stafylopatis, and Stefanos Kollias. Adaptation and contextualization of deep neural network models. In *2017 IEEE Symposium Series on Computational Intelligence (SSCI)*, pages 1–8, 2017.

[14] Weilin Huang, Yu Qiao, and Xiaoou Tang. Robust scene text detection with convolution neural network induced mser trees. In *European Conference on Computer Vision*, pages 497–511. Springer, 2014.

[15] Max Jaderberg, Andrea Vedaldi, and Andrew Zisserman. Deep features for text spotting. In *European conference on computer vision*, pages 512–528. Springer, 2014.

[16] Zhi Tian, Weilin Huang, Tong He, Pan He, and Yu Qiao. Detecting text in natural image with connectionist text proposal network. In *European Conference on Computer Vision*, pages 56–72. Springer, 2016.

[17] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2818–2826, 2016.

[18] Users Guide Matlab. The mathworks. *Inc., Natick, MA*, 1992, 1760.

[19] Franc¸ois Chollet et al. Keras, 2015.

[20] Mart´ın Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. Tensorflow: A system for large-scale machine learning. In *OSDI*, volume 16, pages 265–283, 2016.

[21] Yann LeCun, Leon Bottou, Yoshua Bengio, and Patrick Haffner.´ Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[22] Walmart's food safety solution built on the ibm blockchain platform. https://www.youtube.com/watch?v=SV0KXBxSoio. Accessed: 31-012018.